

CLAIMS:

1. A method for generating a primitive extension defining a generalized primitive, comprising:
 - parameterizing the generalized primitive with parameters;
 - providing an originating primitive; and
 - generating the primitive extension of the originating primitive responsive to the parameters.
2. A method as in claim 1, wherein one of the parameters indicates a number of vertices to form the originating primitive.
3. A method as in claim 1, wherein the generalized primitive is a fan-type primitive.
4. A method as in claim 1, wherein the generalized primitive is a strip-type primitive.
5. A method as in claim 1, wherein one of the parameters indicates a number of new vertices to be added to form an adjacent primitive.
6. A method as in claim 1, wherein one of the parameters indicates a number of vertices of the originating primitive to be used as anchor vertices for each adjacent primitive.
7. A method as in claim 1, wherein the parameters indicate a number of vertices that are shared between two primitives.
8. A method as in claim 6, wherein the parameters indicate a number of vertices in addition to the anchor vertices, needed to define an adjacent primitive.

9. A method as in claim 1, wherein the originating primitive is a point.
10. A method as in claim 1, wherein the originating primitive is a line.
11. A method as in claim 1, wherein the originating primitive is a triangle.
12. A method as in claim 1, wherein the originating primitive is a quadrilateral.
13. A method as in claim 1, wherein the originating primitive is a tetrahedron.
14. A method as in claim 1, wherein the originating primitive is a cube.
15. A method as in claim 1, wherein the originating primitive is a pentagon.
16. A method for generating a primitive extension, comprising:
 - obtaining vertex data for an originating primitive;
 - generating an ordered data stream from the vertex data;
 - selecting a first portion of data from the ordered data stream;
 - skipping a second portion of data immediately past the first portion of data in the ordered data stream; and
 - adding a third portion of data to an end of the ordered data stream.
17. A method as in claim 16, wherein the first portion of the ordered data stream includes at least one anchor vertex, the at least one anchor vertex used to define each primitive within a generalized primitive.
18. A method as in claim 16, wherein the third portion of data includes vertex data for vertices within at least one adjacent primitive.
19. A method as in claim 16, wherein the primitive extension includes vertex data for at least one adjacent primitive, the adjacent primitive including vertex data

from the first portion of data and the third portion of data.

20. A method as in claim 16, wherein the primitive extension is a strip.

21. A method as in claim 16, wherein the primitive extension is a fan.

22. A method as in claim 16, wherein the primitive extension is generated by a primitive engine.

23. A method as in claim 16, further comprising providing the primitive extension to a vertex engine.